(19) BUNDESREPUBLIK DEUTSCHLAND

DEUTSCHES PATENT- UND MARKENAMT

(12) **Offenl gungsschrift**

(10) **DE 100 59 230 A 1**

(51) Int. Cl.[7]: **H 04 L 9/32**

(21) Aktenzeichen: 100 59 230.9
(22) Anmeldetag: 29. 11. 2000
(43) Offenlegungstag: 13. 6. 2002

(71) Anmelder:
4FriendsOnly. com Internet Technologies AG, 98693 Ilmenau, DE

(74) Vertreter:
Engel und Kollegen, 98527 Suhl

(72) Erfinder:
Nützel, Jürgen, Dr.-Ing., 98693 Ilmenau, DE; Böhme, Thomas, Dr.rer.nat. habil., 98693 Ilmenau, DE; Stein, Mathias, 87629 Füssen, DE; Schwetschke, Stefan, 87663 Lengenwang, DE

(56) Entgegenhaltungen:

DE	199 06 450 C1
DE	199 06 449 C1
DE	198 48 492 A1
EP	4 38 154 B1
EP	10 45 386 A1
WO 2	000 64 111 A1
WO 2	000 27 067 A1
WO	99 67 917 A1
JP	10-1 77 523 A1

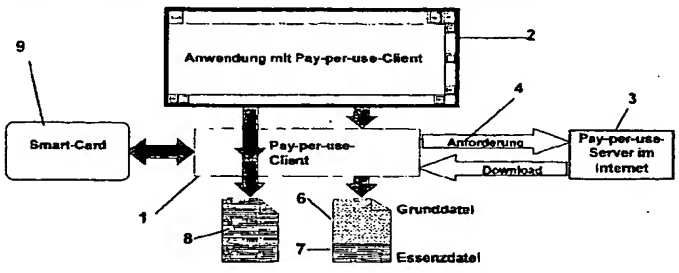
CHENG, H. u.a.: Partial encryption of compressed images and videos. In: IEEE Transactions on Signal Processing, Vol. 48, No. 8, 8. August 2000, S. 2439-2451;

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

Prüfungsantrag gem. § 44 PatG ist gestellt

(54) Verfahren zur Verfügbarmachung von multimedialen Datenmengen und Datenverarbeitungssystem

(57) Die Erfindung betrifft ein Verfahren zur Verfügbarmachung von multimedialen Datenmengen für einen Nutzer. Das Verfahren umfaßt die folgenden Schritte: Bereitstellung einer Essenzdatei (7) auf einem entfernten Server (3), die wenigstens einen Teil der multimedialen Datenmenge umfaßt; Verschlüsseln der Essenzdatei; Übertragung der verschlüsselten Essenzdatei zu einem lokalen Computer (1), auf welchen der Nutzer Zugriff hat; Speicherung der verschlüsselten Essenzdatei auf einem Lokaldatenträger des lokalen Computers; Entschlüsselung der Essenzdatei während der Ausführung eines Datenverarbeitungsprogramms (2) auf dem lokalen Computer; Wiedergabe der multimedialen Datenmenge über ein Ausgabegerät während der Ausführung des Datenverarbeitungsprogramms auf dem lokalen Computer. Vorzugsweise erfolgt dabei eine Aufteilung der multimedialen Datenmenge in eine Grunddatei (6) und eine Essenzdatei (7), gemäß einem vorgegebenen Teilungsalgorithmus, und eine Verknüpfung der Grunddatei und der Essenzdatei zur multimedialen Datenmenge während der Ausführung des Datenverarbeitungsprogramms (2) auf dem lokalen Computer. Die Erfindung betrifft auch ein Datenverarbeitungssystem zur Verfügbarmachung von multimedialen Datenmengen.



PD a 30018

CITED BY APPLICANT

DE 100 59 230 A 1

DE 100 59 230 A 1

Beschreibung

[0001] Die vorliegende Erfindung betrifft ein Verfahren zur Verfügbarmachung von multimedialen Datenmengen für einen oder beliebig viele Nutzer. Darüber hinaus betrifft die Erfindung ein Datenverarbeitungssystem, welches einen Server, einen lokalen Computer und eine Datenübertragungsverbindung zwischen dem Server und dem Computer umfaßt und zur Verfügbarmachung von multimedialen Datenmengen geeignet ist. Unter Verfügbarmachung ist im weiteren Sinne sowohl die Bereitstellung von Daten als auch die Autorisierung des Nutzers zur Nutzung dieser Daten zu verstehen.

[0002] Es sind verschiedene Verfahren bekannt, um elektronische Daten zu vervielfältigen und die hergestellten Kopien an verschiedene Nutzer zu verteilen. Softwareproduzenten sind daran interessiert, ein freies Kopieren bestimmter Daten zu verhindern, um dadurch sicherzustellen, daß investitionsintensive Software in ausreichender Menge verkauft werden kann. Obwohl das unberechtigte Kopieren elektronischer Daten zumindest teilweise durch urheberrechtliche Vorschriften verboten ist, läßt sich die unberechtigte Vervielfältigung von Datenverarbeitungsprogrammen (Anwendungssoftware) und auch von multimedialen Daten verschiedenster Art kaum kontrollieren. Eine Möglichkeit der teilweisen Einschränkung der unberechtigten Vervielfältigung solcher Daten besteht darin, während des Installationsprozesses einer Software einen Daten-Schlüssel abzufragen, der vom Nutzer manuell als Zeichenfolge eingegeben wird und dann die Nutzung der Software ermöglicht. Natürlich läßt sich damit nicht verhindern, daß der zumeist druckschriftlich an den Nutzer übergebene Schlüssel mit den kopierten Daten weitergegeben wird.

[0003] In verschiedenen Bereichen von Anwendungsprogrammen, beispielsweise bei Computerspielen, besteht außerdem das Bedürfnis der Softwarehersteller, die Software möglichst schnell einem breiten Publikum bekanntzumachen, um dadurch die Absatzmenge und -geschwindigkeit zu erhöhen. Dazu werden zeitlich und/oder funktional eingeschränkt lauffähige Versionen der Software kostenlos oder zu einem sehr geringen Preis an potentielle Kunden verteilt. Nachdem der Nutzer diese Software zumindest teilweise testen konnte, besteht die Möglichkeit, eine vollständige Version der Software käuflich zu erwerben. Die elektronische Distribution der eingeschränkten Versionen und der Vollversionen der Software bereitet jedoch Schwierigkeiten, zumindest wenn größere Datenmengen an den Nutzer übergeben werden müssen. Bei der Bereitstellung der Datenmengen auf dauerhaften Datenträgern (z. B. CD-ROM) kommt verschiedentlich ein Verfahren zum Einsatz, bei welchem dem Nutzer nach der Entrichtung des geforderten Kaufpreises lediglich ein Freischaltschlüssel übermittelt wird, der dann den Zugang zu sämtlichen a priori auf dem Datenträger gespeicherten Daten ermöglicht. Dabei besteht jedoch wiederum die Gefahr, daß der Freischaltschlüssel ohne weiteres auch auf angefertigte Kopien der Daten angewendet werden kann.

[0004] Eine Aufgabe der vorliegenden Erfindung besteht somit darin, ein Verfahren zur Verfügbarmachung von insbesondere multimedialen Datenmengen bereitzustellen, durch welches es ermöglicht wird, einem Nutzer ggf. in mehreren Teilschritten Anteile einer multimedialen Datenmenge zu übermitteln, wobei die unkontrollierte Vervielfältigung zumindest von wesentlichen Teilen dieser Datenmenge verhindert werden soll, wobei die Möglichkeit besteht, bestimmte Teile der Datenmenge nur gegen Entrichtung eines Kaufpreises an den Nutzer zu übertragen, und wobei die gegen eine Vergütung bereitgestellte Datenteilmenge auch hinsichtlich ihrer Größe geeignet sein soll, um über Online-Verbindungen übertragen zu werden.

[0005] Diese und weitere Aufgaben werden durch das erfindungsgemäße Verfahren gelöst, welches die folgenden Schritte umfaßt:

- Bereitstellung einer Essenzdatei auf einem entfernten Server, die wenigstens einen Teil der multimedialen Datenmenge umfaßt;
- Verschlüsseln der Essenzdatei;
- Übertragung der verschlüsselten Essenzdatei zu einem lokalen Computer, auf welchen der Nutzer Zugriff hat;
- Speicherung der verschlüsselten Essenzdatei auf einem Lokaldatenträger des lokalen Computers;
- Entschlüsselung der Essenzdatei während der Ausführung eines Datenverarbeitungsprogramms auf dem lokalen Computer;
- Wiedergabe der multimedialen Datenmenge über ein Ausgabegerät, während der Ausführung des Datenverarbeitungsprogramms auf dem lokalen Computer.

[0006] Dieses Verfahren ermöglicht es ganz allgemein, multimediale Daten beliebig vielen Nutzern verfügbar zu machen, wobei einerseits die zum Zugriff auf die Daten berechtigten Nutzer nach bestimmten Kriterien ausgewählt werden können (z. B. Zahlung von Entgelten, Registrierung oder dergleichen) und andererseits eine ungehinderte Vervielfältigung und Verbreitung der Daten sicher vermieden werden kann. Überall wo multimediale Daten wesentliche Bestandteile von Softwareanwendungen sind, kann das Verfahren eingesetzt werden. Ein signifikanter Unterschied beispielsweise zu herkömmlichen Verfahren, bei denen lediglich eine Freischaltung von dem potentiellen Nutzer bereits übergebenen Daten erfolgt, fehlen dem Nutzer hier die multimedialen Daten oder zumindest wesentliche Teile dieser Daten, so daß auch bei Kenntnis eines Freischaltschlüssels die Softwareanwendungen nicht oder nur mit erheblichen Einschränkungen betrieben werden können, solange nicht von einem autorisierten Verteiler die benötigten Daten bezogen werden.

[0007] Gemäß einer besonders zweckmäßigen Ausführungsform umfaßt das Verfahren weiterhin die folgenden Schritte:

- Aufteilung der multimedialen Datenmenge in eine Grunddatei und eine Essenzdatei, gemäß einem vorgegebenen Teilungsalgorithmus;
- Übertragung der Grunddatei und der Essenzdatei;
- Speicherung der Grunddatei und der verschlüsselten Essenzdatei auf einem Lokaldatenträger eines lokalen Computers;
- Verknüpfung der Grunddatei und der Essenzdatei zur multimedialen Datenmenge während der Ausführung des Datenverarbeitungsprogramms auf dem lokalen Computer, nach der Entschlüsselung der Essenzdatei.

[0008] Diese Variante des Verfahrens ermöglicht es, eine oder mehrere Dateien der multimediale Datenmenge in jeweils zwei Dateien, eine sogenannte Grunddatei und eine (oder mehrere) sogenannte Essenzdateien aufzuspalten. Im Unterschied zu der o. g. Ausführungsform liegen dem Nutzer mit der Grunddatei bereits Teile der multimedialen Datenmenge vor, die jedoch erst beim Vorhandensein der Essenzdatei nutzbar oder zumindest erst dann vollwertig nutzbar werden (z. B. hinsichtlich Qualität und Funktion). Die Teildateien, die in diesem Fall Teile einer Gesamtdatei sind

(der multimedialen Datenmenge), können auf unterschiedlichen Datenübertragungskanälen an interessierte Nutzer verteilt werden. Dabei ist es vorteilhaft, daß die Grunddatei wesentlich größer als die Essenzdatei sein kann, so daß die Grunddatei beispielsweise auf einem dauerhaften und preiswerten Datenträger, beispielsweise einer CD-ROM, dem Nutzer übermittelt werden kann. Die Essenzdatei wird hinsichtlich des Dateninhalts so ausgewählt, daß sie in den meisten Fällen zwar deutlich kleiner als die Grunddatei ist, jedoch wesentliche Informationen enthält, die für eine qualitativ hochwertige und/oder vollständige Wiedergabe der multimedialen Datenmenge von entscheidender Bedeutung sind. Die Essenzdatei kann vorzugsweise durch eine Online-Übertragung auf gesonderte Anforderung des Nutzers zu diesem übertragen werden. Damit ist es möglich, dem Empfänger beliebig viele Essenzdateien ggf. gegen zu entrichtende Vergütungen auf Wunsch unmittelbar zuzuleiten.

[0009] Die Verschlüsselung der Essenzdatei bringt außerdem den Vorteil mit sich, daß die Weitergabe der verschlüsselten Essenzdatei an unberechtigte Nutzer nicht erfolgen wird, da diese ohne Kenntnis des privaten Schlüssels des berechtigten Nutzers zur Entschlüsselung der Essenzdatei nicht in der Lage sind.

[0010] Die Erfindung stellt außerdem ein Datenverarbeitungssystem der o. g. Art bereit, welches weiter umfaßt:

- eine Verschlüsselungseinrichtung, die eine Essenzdatei verschlüsselt, welche wenigstens einen Teil der multimedialen Datenmenge umfaßt;
- eine Archiverzeugungseinrichtung des Servers, die die verschlüsselte Essenzdatei mit nutzerspezifischen Daten zu einem Transferarchiv kombiniert und diese für die Online-Übertragung über die Datenübertragungsverbindung bereitstellt;
- eine Aktualisierungseinrichtung des lokalen Computers, die Einträge in eine Systemregistrierungsdatenbank des lokalen Computers vornimmt, welche die Inhalte des übertragenen Transferarchivs angeben;
- eine Entschlüsselungseinrichtung des lokalen Computers, welche während der Ausführung eines Datenverarbeitungsprogramms die Essenzdatei entschlüsselt;
- eine Ausgabeeinrichtung, welche die multimediale Datenmenge an das Ausgabegerät ausgibt.

[0011] Vorzugsweise sind die genannten Einrichtungen des Datenverarbeitungssystems durch geeignete Softwareroutinen realisiert, so daß hinsichtlich ihrer gerätetechnischen Merkmale bzw. ihrer Hardware herkömmliche Computer eingesetzt werden können.

[0012] Die Erfindung stellt auch geeignet konfigurierte Computer bereit, die die Aufgaben des Servers bzw. des lokalen Computers übernehmen können, um gemeinsam die Schritte des erfindungsgemäßen Verfahrens auszuführen.

[0013] Gemäß einer zweckmäßigen Ausführungsform des Verfahrens erfolgt die Verschlüsselung der Essenzdatei in einem ersten Schritt (bei Anforderung, wenn Archiv erstellt wird) unter Anwendung eines symmetrischen Verschlüsselungsalgorithmus. Vorzugsweise wird die Verschlüsselung erst ausgeführt, wenn die Essenzdatei von einem Nutzer angefordert wird, wobei bei jeder Sitzung/Anforderung ein sitzungsspezifischer Schlüssel erzeugt wird (Session-Key). Daher erhält jeder Nutzer die Essenzdatei mit einem anderen Schlüssel verschlüsselt. Der dafür erforderliche sitzungsspezifische Schlüssel, der bei Bedarf beispielsweise als Zufallszahl "gewürfelt" wird, wird in einem zweiten Schritt in einem asymmetrischen Verschlüsselungsalgorithmus verschlüsselt, wobei dabei der öffentliche Schlüssel des Nutzers verwendet wird, welcher die Übersendung der Essenzdatei angefordert hat.

[0014] Gemäß einer besonders bevorzugten Ausführungsform kann der mit dem öffentlichen Schlüssel des Nutzers verschlüsselte Session-Key zusätzlich mit einem nutzerspezifischen Schlüssel (Passwort) symmetrisch verschlüsselt werden. Damit kann eine Identifizierung des Nutzers auch in Fällen gewährleistet werden, wenn der Nutzer nicht mit einer Smart-Card ausgerüstet ist. Das Passwort kann dem Nutzer beispielsweise per email übermittelt werden, wodurch die email Adresse überprüft wird.

[0015] Da die Entschlüsselung der Essenzdatei erst während der Abarbeitung des Datenverarbeitungsprogramms auf dem lokalen Computer erfolgt, ist eine Speicherung der Essenzdatei im unverschlüsselten Zustand auf dem lokalen Computer zu keinem Zeitpunkt erforderlich. Für einen unberechtigten Dritten ist der Zugriff auf die verschlüsselte Essenzdatei damit nutzlos, da er den zur Entschlüsselung der Essenzdatei erforderlichen privaten Schlüssel des berechtigten Nutzers nicht kennt.

[0016] Gemäß einer vorteilhaften Ausführungsform des durch die Erfindung bereitgestellten Datenverarbeitungssystems wird der private Schlüssel des Empfängers auf einer Smart-Card aufbewahrt, aus welcher dieser private Schlüssel nicht ausgelesen werden kann. Bei der Entschlüsselung der Essenzdatei bzw. des symmetrischen Schlüssels greift das Datenverarbeitungsprogramm auf die Smart-Card zu, um die asymmetrische Entschlüsselung des symmetrischen sitzungsspezifischen Schlüssels (Session-Key) vorzunehmen. Für den Fall, daß der private Schlüssel zusätzlich mit einem Passwort verschlüsselt war, muß zuvor die Entschlüsselung mit dem Passwort erfolgen.

[0017] Gemäß einer bevorzugten Ausführungsform des Verfahrens wird auf dem Server ein Transferarchiv erzeugt, welches die verschlüsselte Essenzdatei, eine Serversignatur, einen Prüfabschnitt, einen Datenidentifikationsabschnitt und einen Nutzerdatenabschnitt enthält. Die Serversignatur ermöglicht es dem Empfänger, die Identität des Servers zu überprüfen, so daß verhindert wird, daß unberechtigte Dritte Funktionen des Servers übernehmen und dabei beispielsweise unberechtigt Zahlungen vom Empfänger entgegennehmen. Der Datenidentifikationsabschnitt enthält u. a. alle benötigten Angaben, die zum Zusammensetzen der Grunddatei und der Essenzdatei erforderlich sind. Beispielsweise ist es möglich, daß im Transferarchiv ein Verweis auf einen Algorithmus enthalten ist, der bereits beim Nutzer/Empfänger als Teil des Datenverarbeitungsprogramms vorliegt und die korrekte Zusammensetzung von Grunddatei und Essenzdatei ermöglicht. Es ist aber auch denkbar, daß das Transferarchiv den benötigten Algorithmus zur Zusammensetzung dieser Dateien selbst enthält.

[0018] Der Nutzerdatenabschnitt des Transferarchivs ermöglicht die Identifizierung des Nutzers/Empfängers. Entsprechende Daten werden bei Bedarf auf Server in einer speziellen Datenbank gespeichert. Auf diese Weise kann zu einem späteren Zeitpunkt nachvollzogen werden, welcher Nutzer welches Transferarchiv und damit welche Essenzdatei erhalten hat. Diese Angaben sind erforderlich, um eine korrekte Abrechnung durchzuführen, wenn die Essenzdatei gegen Entgelte an den Empfänger gesandt wird. Außerdem können bereits vollzogene Übertragungen wiederholt werden, wenn dies aufgrund von Übertragungsstörungen oder anderen Datenverlusten erforderlich wird.

[0019] Die Handhabung der auf den lokalen Computer übertragenen verschlüsselten Essenzdatei wird vereinfacht, indem in einer Systemregistrierungsdatenbank des lokalen Computers entsprechende Einträge generiert werden. Aus solchen Einträgen kann das ablaufende Datenverarbeitungs-

programm die Speicherposition des Transferarchivs sowie der darin enthaltenen Essenzdateien ermitteln, um auf diese zuzugreifen. Die Systemregistrierungsdatenbank enthält damit Angaben darüber, welche Essenzdateien auf dem lokalen Computer vorhanden sind und in welchem Transferarchiv sie gespeichert sind. Bei fehlenden Essenzdateien kann eine Online-Verbindung zum Server aufgebaut werden, um die benötigte Essenzdatei übertragen zu lassen.

[0020] Weitere Vorteile, Einzelheiten und Weiterbildungen ergeben sich aus der nachfolgenden Beschreibung bevorzugter Ausführungsformen der Erfindung unter Bezugnahme auf die Zeichnung. Es zeigen:

[0021] **Fig.** 1 ein Blockschaltbild eines erfindungsgemäßen Datenverarbeitungssystems;

[0022] **Fig.** 2 eine Prinzipdarstellung der Zusammensetzung einer Grunddatei und einer Essenzdatei zu einer multimedialen Datenmenge;

[0023] **Fig.** 3 ein Ablaufdiagramm, welches die wichtigsten Verfahrensschritte bei der Anforderung und Übertragung einer Essenzdatei zeigt;

[0024] **Fig.** 4 die allgemeine Struktur einer Formularseite, die während der Anforderung einer Essenzdatei angewendet wird;

[0025] **Fig.** 5 die allgemeine Struktur eines Transferarchivs.

[0026] **Fig.** 1 zeigt in einem Blockdiagramm den generellen Aufbau eines erfindungsgemäßen Datenverarbeitungssystems. Im folgenden soll dieses System als Pay-per-use-Anwendung bezeichnet werden. Pay-per-use soll dabei als schrittweiser Verkauf einer multimedialen Datenmenge verstanden werden.

[0027] Auf einem lokalen Computer **1** wird ein Datenverarbeitungsprogramm **2** (Anwendungssoftware) ausgeführt, im dargestellten Beispiel ein PC-Spiel, wodurch der lokale Computer als Client konfiguriert wird. Weiterhin existiert ein Server **3**, der vorzugsweise als Internet-Server aufgebaut ist und auf den der lokale Computer **1** über eine zeitweilige Online-Verbindung **4** zugreifen kann. Über die Online-Verbindung **4** wird sowohl eine Datenanforderung vom lokalen Computer zum Server übertragen als auch ein sogenannter Download durchgeführt, d. h. eine Datenübertragung vom Server zum lokalen Computer.

[0028] Bei der Ausführung des Datenverarbeitungsprogramms **2** greift der lokale Computer **1** auf eine Grunddatei **6** zu. Die Grunddatei steht in diesem Fall bereits auf einem dauerhaften Datenträger zur Verfügung. Beispielsweise erhält der Nutzer eine CD-ROM, die u. a. die Grunddatei **6** umfaßt und dem lokalen Computer bereitgestellt wird. Bei der Grunddatei **6** handelt es sich um eine Teildatei aus einer multimedialen Datenmenge, die dem lokalen Computer anfänglich nicht vollständig zur Verfügung steht. Die Differenz zwischen der multimedialen Datenmenge und der Grunddatei **6** ist eine Essenzdatei **7**, die für eine vollwertige Ausführung des Datenverarbeitungsprogramms **2** ebenfalls erforderlich ist. Die Essenzdatei **7** liegt auf dem Server **3** bereit und kann als eine Ergänzung der Grunddatei (Plugin) verstanden werden. Die Essenzdatei **7** wird bei der entsprechenden Berechtigung vom Server **3** geladen. Vor der Übertragung erfolgt jedoch eine Verschlüsselung der Essenzdatei **7**. Die Einzelheiten der Zerlegung der multimedialen Datenmenge in die Grunddatei und Essenzdatei sowie die Möglichkeiten des Erhalts und der Speicherung der Essenzdatei werden weiter unten detailliert erläutert.

[0029] Bei einer abgewandelten Ausführungsform besteht die vom Anwendungsprogramm benötigte multimediale Datenmenge nicht aus Teildateien sondern ist in einer vollständigen Essenzdatei **8** abgebildet, d. h. es existiert keine Grunddatei. Damit ist es beispielsweise möglich, funktionale Erweiterungen für das Anwendungsprogramm bereitzustellen, die zum Zeitpunkt der Lieferung des Anwendungsprogramms an den Nutzer ggf. noch nicht zur Verfügung standen. Die vollständige Essenzdatei **8** könnte z. B. ein neues Level eines PC-Spiels enthalten, für welches beim Anwender noch keine Grunddatei existiert.

[0030] Die in **Fig.** 1 gezeigte Ausführungsform umfaßt weiterhin eine Smart-Card-Einrichtung **9**, die der Verschlüsselung bzw. Entschlüsselung von Daten dient. Die Smart-Card-Einrichtung ist in der Lage, mit einer Smart-Card zusammenzuarbeiten, auf welcher ein privater Schlüssel des Benutzers/Empfängers gespeichert ist. Bei anderen Ausführungsformen kann die Smart-Card-Einrichtung **9** durch eine "virtuelle Smart-Card" ersetzt sein, d. h. der private Schlüssel ist auf einem lokalen Datenträger des lokalen Computers **1** abgelegt bzw. in besonderer Weise versteckt (z. B. mit Hardware-Parametern verschlüsselt). Die Einzelheiten der Verschlüsselung bzw. Entschlüsselung von Daten, soweit sie dem Fachmann nicht ohnehin bekannt sind, werden ebenfalls weiter unten detailliert erläutert.

[0031] Um das hier zur Anwendung kommende Prinzip der Datenverteilung, -übertragung und -zusammensetzung für mehrere Datenverarbeitungsprogramme nutzen zu können wird durch geeignete Softwareroutinen auf dem lokalen Computer **1** ein Pay-per-use-Client abgebildet. Der Pay-per-use-Client kann beispielsweise eine dynamisch ladbare Programmbibliothek (DLL) sein. Dabei ist es ausreichend, wenn die Programmierschnittstelle (API) des Clients offengelegt wird, damit verschiedene Anwendungsprogramme auf diese Schnittstelle zugreifen können. Die Bekanntgabe des Quellcodes des Pay-per-use-Clients ist nicht erforderlich, was auch aus Sicherheitsgründen wünschenswert ist.

[0032] Das Datenverarbeitungsprogramm (Anwendungssoftware) kann beispielsweise als Windows-Anwendung realisiert sein und die unter diesem Betriebssystem gängigen Programmiermöglichkeiten nutzen. Das Betriebssystem soll zumindest in der Lage sein, die Verwaltung von Dateien, d. h. Grunddatei und Essenzdatei, durchführen zu können, d. h. ein geeignetes Dateisystem muß vorhanden sein. Diese Selbstverwaltung umfaßt die Fähigkeit, die Dateien von einem Datenträger (z. B. Festplatte) zu lesen und zur Abarbeitung in den Hauptspeicher zu übertragen.

[0033] Das Ziel des Aufteilens von multimedialen Datenmengen in zumindest eine Grunddatei und zumindest eine Essenzdatei besteht vor allem darin, die entstehenden Teildateien auf unterschiedlichen Distributionskanälen an potentielle Nutzer/Empfänger zu übermitteln. Generell ist eine Aufteilung einer Datenmenge auch bei anderen Datenformen möglich, jedoch würde beispielsweise die Entfernung eines bestimmten Anteils von Daten aus einer ausführbaren Datei (Anwendungsprogramm) wenig sinnvoll sein, wenn angestrebt ist, daß die verbleibende Grunddatei beispielsweise zu Demonstrationszwecken weiterhin ausführbar ist. Wenn jedoch bei der Ausführung eines Datenverarbeitungsprogramms multimediale Datenmengen benötigt werden, kann das Herausschneiden eines bestimmten Anteils dieser Datenmengen so erfolgen, daß das Datenverarbeitungsprogramm weiterhin ausführbar bleibt, wenngleich mit eingeschränkter Qualität oder verringertem Funktionsumfang. Wie oben bereits erläutert wurde, können auch multimediale Datenmengen bereitgestellt werden, die nicht zerteilt sind. Die Aufteilung der Datenmenge ist insbesondere bei multimedialen Daten von Interesse, da beispielsweise Bild- und Tondaten auch bei der Anwendung von Komprimierungsverfahren zu relativ großen Dateien führen, die beispielsweise über Online-Verbindungen nicht mehr oder nur mit hohem Kosten- und Zeitaufwand übertragen werden können. Die hier verwendete Aufteilung der Datenmengen ist

somit besonders für multimediale Daten geeignet. Dabei wird angestrebt, daß die Grunddatei hinsichtlich ihrer Qualität und/oder ihres Funktionsumfangs soweit eingeschränkt ist, daß eine sinnvolle und vollwertige Nutzung nicht mehr möglich ist. Aus der Gesamtmenge der multimedialen Daten müssen daher in Abhängigkeit von der Art der Daten diejenigen Daten in eine oder mehrere Essenzdateien verlagert werden, die einen solchen Qualitäts- oder Funktionsverlust bewirken. Andererseits besteht die Forderung, die Essenzdatei möglichst klein zu halten, um insbesondere eine Online-Übertragung der Essenzdatei zu ermöglichen. Nachfolgend werden Beispiele genannt, die für spezielle Datenformate die Aufteilung in die Grunddatei und die Essenzdatei verdeutlichen.

[0034] In Computerspielen aber auch bei CAD- und anderen Planungsanwendungen werden häufig Bilder oder dreidimensionale Modelle eingesetzt, um am Bildschirm bestimmte Szenarien darzustellen. Diese multimedialen Daten werden von der jeweiligen Anwendungssoftware jeweils vollständig benötigt, um eine exakte grafische Anzeige zu generieren. Bei Dateien, die solche Daten enthalten, sind häufig am Dateianfang (Header) notwendige Informationen zur Interpretation der nachfolgenden Daten abgespeichert (z. B. GIF-Dateien). Wenn dieser Dateikopf entfernt wird, fehlen notwendige Daten, um die übrigen in dieser Datei enthaltenen Daten korrekt zur Anzeige zu bringen. Beispielsweise kann bei einer farbigen GIF-Grafik die Farbtabelle (3 × 256 Bytes) entfernt werden, wodurch die gesamte Datei weitgehend unbrauchbar wird. Die gewünschte Aufteilung der multimedialen Datenmenge wird also bei solchen Datenformaten derart erfolgen, daß die Grunddatei den größten Teil der Datenmenge enthält, während die Essenzdatei lediglich den Dateikopf oder bestimmte Informationen aus dem Dateikopf umfaßt.

[0035] Andere Daten besitzen einen Dateityp, der als "dynamisch" bezeichnet werden kann. Hierbei handelt es sich beispielsweise um Video- oder Audiodateien, wobei die aneinandergereihten Daten in einer zeitlich definierten Beziehung zueinander stehen. Bei diesen Dateitypen wäre es nicht sinnvoll, nur einige Anfangswerte der Dateien herauszuschneiden, da dann alle zeitlich späteren Daten ungestört wiedergegeben werden könnten. Vielmehr ist es erforderlich, die multimediale Datenmenge zeitlich kontinuierlich zu zerlegen, so daß wesentliche Teile der Daten über den gesamten Zeitraum der Wiedergabe der Gesamtdatenmenge in die Essenzdatei verlagert werden. Die multimediale Datenmenge wird in diesem Fall kontinuierlich wiedergegeben, wobei die jeweilige Software kleinere Datenpakete nacheinander verarbeitet und zur Wiedergabe bringt. Wenn beispielsweise ein stereophones Audiosignal die multimediale Datenmenge darstellt, kann eine Zerlegung derart erfolgen, daß nur die Stereokomponente (Differenz aus rechtem und linkem Kanal) in die Essenzdatei verlagert wird. Die Grunddatei enthält dann das monophone Audiosignal, so daß für eine qualitativ hochwertige Stereowiedergabe die Essenzdatei benötigt wird. Wenn die Datenmenge eine Videodatei ist, könnten beispielsweise alle 10 Sekunden vollständige Bilder in die Essenzdatei verlagert werden. Wenn zur Verarbeitung von Videodaten Deltakommpressionsverfahren eingesetzt werden, bei denen immer nur der Unterschied zum vorherigen Bild gespeichert wird, ist es besser jeweils das erste Bild nach einem Szenenwechsel in die Essenzdatei zu verlagern.

[0036] Generell sollte darauf geachtet werden, daß bei der Zerteilung der multimedialen Datenmenge keine redundanten Daten in die Essenzdatei verlagert werden, da diese ohne weiteres in der Grunddatei reproduziert werden können, ohne daß dafür die Essenzdatei erforderlich ist. Bei bestimmten Dateitypen kann es daher zweckdienlich sein, die Zerlegung der Datenmenge erst nach einer geeigneten Komprimierung durchzuführen, durch welche die redundanten Daten weitgehend ausgeblendet werden.

[0037] **Fig.** 2 zeigt in einem Blockdiagramm das Prinzip der Verarbeitung von multimedialen Datenmengen, die zerteilt in die Grunddatei **6** und die Essenzdatei **7** vorliegen. Bei dem dargestellten Beispiel handelt es sich um "dynamische" Datentypen, die auch als Streaming-Daten bezeichnet werden. Die multimediale Datenmenge umfaßt beispielsweise 100 Datenpakete. Aus der Gesamtdatenmenge werden im gezeigten Fall zehn Datenpakete entnommen und in die Essenzdatei **7** verlagert. Der Grunddatei **6** fehlen damit (mehr oder wenjger gleichmäßig) über die Datenmenge verteilt wesentliche Daten, die zur Wiedergabe der multimedialen Datenmenge in hoher Qualität erforderlich sind.

[0038] Bei der Zusammensetzung der Grunddatei **6** und der Essenzdatei **7** auf dem lokalen Computer erfolgt eine Entschlüsselung der Datenpakete der Essenzdatei **7** in einer Entschlüsselungseinrichtung **11**, da die Essenzdatei **7** nur in verschlüsseltem Zustand auf dem lokalen Computer vorliegt (siehe unten). In einem Mischer **12** werden anschließend die einzelnen Datenpakete aus der Grunddatei **6** und der Essenzdatei **7** in der richtigen Reihenfolge zusammengefügt. Dem Mischer **11** muß dazu die Vorschrift zum Zusammensetzen der Daten bekannt gegeben werden. Die dann wieder in der richtigen Reihenfolge vorliegenden Datenpakete werden in herkömmlicher Weise einem Decoder **13** eingespeist, der ein multimediales Ausgangssignal generiert und es einem Ausgabegerät **14** zuleitet.

[0039] Die Art und Weise der Zusammensetzung der Grunddatei und der Essenzdatei hängt von den verwendeten Datentypen und dem zur Datenaufteilung eingesetzten Verfahren ab. Dem auf dem lokalen Computer ablaufenden Anwendungsprogramm müssen daher Informationen über den zur verwendenden Algorithmus zur Zusammensetzung der Grunddatei und der Essenzdatei übermittelt werden. Einzelheiten dazu werden weiter unter erläutert.

[0040] **Fig.** 3 zeigt ein Ablaufdiagramm, in welchem einzelne Schritte dargestellt sind, die vor bzw. während der Übertragung einer Essenzdatei von einem Server zu einem lokalen Computer ausgeführt werden. Wie oben bereits erläutert wurde, wird auf dem lokalen Computer ein Datenverarbeitungsprogramm ausgeführt, welches eine multimediale Datenmenge benötigt. Wenn das Datenverarbeitungsprogramm eine bestimmte Funktion abarbeiten will, wird durch eine Abfrage einer lokalen Systemregistrierungsdatenbank ermittelt, ob die benötigte Essenzdatei **7** auf dem lokalen Computer vorhanden ist, die zur Ergänzung der Grunddatei **6** erforderlich ist. Unter dem Betriebssystem Windows ist die Systemregistrierungsdatenbank durch die sogenannte Registry (RG) bereitgestellt. Die Registry enthält auch Einträge, aus denen das Anwendungsprogramm ermitteln kann, ob der entsprechende Benutzer erstmalig auf die erforderliche Essenzdatei zugreifen will, ob ein wiederholter Zugriff erfolgt oder ob für das gleiche Datenverarbeitungsprogramm bereits andere Essenzdateien auf dem lokalen Computer vorhanden sind.

[0041] In Bezug auf **Fig.** 3 wird zuerst der Fall betrachtet, daß das Datenverarbeitungsprogramm erstmalig auf eine Funktion zugreift, welche eine bestimmte Essenzdatei benötigt. Dieser Fall wird hier als Neuanmeldung bezeichnet, wobei der Benutzer des lokalen Computers ein Neukunde (Newcustomer) ist. Das Verfahren startet im Schritt **20**. Der Benutzer erhält im Schritt **21** den Hinweis, daß die benötigte Essenzdatei nicht auf dem lokalen Computer vorhanden ist und daher eine Online-Verbindung mit einem Pay-per-use-Server hergestellt werden kann, um die Essenzdatei von dort

zu erhalten.

[0042] Der Pay-per-use-Server ist ein Rechner, der eine permanente Verbindung zum Internet besitzt. Die auf diesem Server ablaufende Software ist mit einem sogenannten E-Commerce-Shop vergleichbar. Der Server besitzt eine Datenschnittstelle zu dem angeschlossenen Datennetz, im erläuterten Beispiel also zum World Wide Web. Der Server ist jedoch insoweit besonders konfiguriert, daß er nur durch einen speziellen Pay-per-use-Client angesprochen werden kann. Wie oben dargestellt wurde, ist der Pay-per-use-Client auf dem lokalen Computer realisiert. Die Beschränkung der Zugriffsmöglichkeiten auf den Server erhöht die Sicherheit, da beliebige andere Rechner nicht ohne weiteres auf diesen Server zugreifen können. In **Fig.** 3 ist dieser speziell eingerichtete Server mit der Bezeichnung "4fo" gekennzeichnet.

[0043] Wenn sich der Benutzer im Schritt **21** entscheidet, eine Verbindung mit dem Server herzustellen, wird im Schritt **22** die benötigte Verbindung zu 4fo hergestellt und die einleitende Datenübertragung kann beginnen. Der lokale Computer könnte aber auch so konfiguriert werden, daß kein Eingriff des Benutzers erforderlich ist sondern die jeweils benötigte Essenzdatei bei Bedarf automatisch vom Server geladen wird. Dazu übermittelt der Client spezielle Parameter an den Server, unter Nutzung der sogenannten POST-Methode. Diese Parameter sind erforderlich, um dem Server mitzuteilen, welche Informationen (Essenzdateien) benötigt werden.

[0044] Um die Identität des Benutzer bzw. lokalen Computers festzustellen, werden die übertragenen Parameter zusätzlich digital signiert. Dazu wird vorab beispielsweise durch Anwendung eines asymmetrischen Verschlüsselungsverfahrens ein Schlüsselpaar gebildet. Der öffentliche Teil des Schlüsselpaars (Public-Key) wird an den Server übertragen. Sofern wie in Bezug auf **Fig.** 1 erläutert wurde, eine Smart-Card eingesetzt wird, kann der Public-Key aus der Smart-Card ausgelesen werden.

[0045] Um auch eine Autorisierung des Servers gegenüber dem Client zu ermöglichen, wird ein zweites Schlüsselpaar verwendet. Dieses zweite Schlüsselpaar ist anwendungsspezifisch. Der privtae Schlüssel verbleibt auf dem Server. Der öffentliche Schlüssel ist Bestandteil des Clients. Wenn die vom Server übertragenen Daten unter Nutzung des Private-Key signiert werden, kann durch den lokalen Computer überprüft werden, ob die empfangenen Daten tatsächlich von dem angewählten Server stammen, so daß eine Datenmanipulation durch Dritte ausgeschlossen ist.

[0046] Im Schritt **23** überprüft der Server die Signatur des Nutzers, der sich erstmalig anmeldet. Wenn im Schritt **24** festgestellt wird, daß die Parameter unverfälscht vom neuen Nutzer übertragen wurden, beginnt im Schritt **25** die Registrierung des Neukunden. Dazu wird für den Neukunden ein Login (Benutzername) festgelegt.

[0047] **Fig.** 4 zeigt die allgemeine Struktur einer möglichen Formularseite, die bei der Anmeldung eines Benutzers am Server Verwendung findet. Diese Formularseite besitzt eine konstante Menüleiste und einen veränderlichen Teil. Natürlich kann bei abgewandelten Ausführungsformen auch ein anderer Aufbau gewählt werden.

[0048] Zurück zu **Fig.** 3. Dort ist in einzelnen Verfahrensschritten die Formularseite gemäß **Fig.** 4 erkennbar, wobei bestimmte Steuerungsfelder ausgeblendet sind, wenn die zugeordneten Aktionen an der speziellen Stelle im Verfahrensablauf nicht zur Verfügung stehen. Zur Überprüfung der Daten des Benutzer kann im Schritt **26** die Mitteilung des Login und insbesondere des Passwortes per E-Mail an den Benutzer erfolgen. Damit ist zumindest sichergestellt, daß die E-Mail-Adresse des Nutzers korrekt ist. Nachdem im Schritt **27** die Datenüberprüfung als erfolgreich bestätigt wurde, werden im Schritt **28** ein Initialisierungspasswort und eine Kundennummer ermittelt. Das Passwort ist beispielsweise eine 64-Bit Zahl, die Base64 codiert wurde. Das codierte Passwort und die Kundennummer werden per E-Mail im Schritt **29** an den Nutzer übermittelt. Das Passwort kann vom Nutzer auf Wunsch lokal gespeichert werden oder jedesmal bei der Ausführung des Datenverarbeitungsprogramms eingegeben werden. Die Kundennummer wird benutzt, wenn Zahlungen für den Erhalt der Essenzdateien zu leisten sind und diese Zahlungen durch telefonische Anweisungen ausgelöst werden sollen.

[0049] Nachfolgend wird auf dem Server im Schritt **30** ein Transferarchiv erzeugt, welches die Essenzdatei in verschlüsselter Form enthält. Vorzugsweise wird die Essenzdatei jedesmal mit einem einzigartigen sitzungsspezifischen Schlüssel (Session-Key) unmittelbar vor der Einbindung in das Transferarchiv verschlüsselt. Der spezielle Aufbau des Transferarchivs wird weiter unter erläutert. Im Schritt **31** wird das erstellte Transferarchiv zum lokalen Computer übertragen. Dazu wird ein spezielle Komponente verwendet die es ermöglicht, daß Transferarchiv entgegen zu nehmen und in den Hauptspeicher des lokalen Computers zu übergeben. Sobald das Transferarchiv auf dem lokalen Computer vorhanden ist, wird es auf Unversehrtheit und korrekte digitale Signatur überprüft und kann dann auf einem lokalen Datenträger gespeichert werden. Bei der Erstellung des Transferarchivs auf dem Server kann ein eindeutiges Sitzungsmerkmal erzeugt und im Transferarchiv abgelegt werden, aus welchem ein eindeutiger Dateiname zur Speicherung auf dem lokalen Computer generiert werden kann. Dazu kann z. B. die Generierungszeit des Transferarchivs verwendet werden. Außerdem enthält das Transferarchiv den o. g. Session-Key in verschlüsselter Form.

[0050] Schließlich erfolgt im Schritt **32** die Installation des Transferarchivs beim Nutzer, in dem u. a. das Transferarchiv und die darin enthaltenen Essenzdateien in der Systemregistrierungsdatenbank angemeldet werden. Damit ist es auch möglich, bei erneuter Übertragung bestimmter Essenzdateien (z. B. im Falle einer Aktualisierung) den Speicherort der jeweiligen Essenzdateien in der Registrierungsdatenbank zu aktualisieren, so daß das Datenverarbeitungsprogramm jeweils auf die aktuellste Fassung der einzelnen Essenzdateien zugreift. An dieser Stelle könnte das Verfahren enden, da die Essenzdatei auf dem lokalen Computer installiert wurde und für den berechtigten Nutzer nutzbar ist. Das Verfahren kann aber auch fortgesetzt werden, wie es in **Fig.** 3 ersichtlich ist. Die sich anschließenden Verfahrensschritte werden weiter unten erläutert.

[0051] **Fig.** 5 zeigt die allgemeine Struktur eines Transferarchivs, wie es auf dem Server erstellt wird. Generell besteht das Transferarchiv aus einer oder mehreren Essenzdateien (Plugins), einem Archivkopf (ArchiveHeader), einer vom Server bereitgestellten digitalen Signatur (serverSignature) und einer Prüfsumme (Crc32). Vor der Einbindung der Essenzdatei in das Transferarchiv wir diese unter Anwendung eines symmetrischen Verschlüsselungsverfahrens verschlüsselt. Den symmetrischen Schlüssel bildet z. B. eine 128-Bit Zahl, die als Sitzungsschlüssel (Session-Key) auf dem Server für jede einzelne Sitzung bzw. Anforderung durch einen Nutzer zufällig ermittelt wird. Dieser Sitzungsschlüssel wird weiterhin mit dem Public-Key des jeweiligen Nutzers asymmetrisch verschlüsselt und anschließend ebenfalls im Transferarchiv abgelegt. Die Entschlüsselung der Essenzdatei 7 kann somit nur erfolgen, wenn der Private-Key des Nutzers vorhanden ist, der beispielsweise in der Smart-Card abgelegt ist. Die Prüfsumme Crc32, die ebenfalls in das Transferarchiv eingebunden ist, kann verwendet werden, um zu überprüfen, ob die Entschlüsselung korrekt

ausgeführt wurde. Eine weitere Erhöhung der Sicherheit wird erzielt, indem der private Schlüssel des am Server angemeldeten Schlüssels zusätzlich symmetrisch mit dem vergebenen Passwort verschlüsselt wird, welches zur Überprüfung der Identität des Nutzers per E-Mail an diesen gesandt werden kann.

[0052] Bei einer praktischen Ausführung können die folgenden bekannten Algorithmen zur Verschlüsselung eingesetzt werden:

Asymmetrisches Verfahren: RSA 1024 Bit

Symmetrisches Verfahren: Blowfish 128 Bit (Safer, Square)

Passwort: 64-Bit Zufallszahl, die mit Base64 umkodiert wurde

Hashfunktion (für digitale Signatur): SHA1

[0053] Das Datenverarbeitungsprogramm, welches auf dem lokalen Computer abläuft, ruft beim Zugriff auf die jeweilige Essenzdatei die Entschlüsselungsroutine auf, so daß die Entschlüsselung in Echtzeit erfolgt und die Essenzdatei nur im verschlüsselten Zustand auf den permanenten Datenträgern des lokalen Computers vorhanden ist. Bei Bedarf kann die Essenzdatei vor dem Verschlüsseln verlustfrei komprimiert werden, was eine Dekompression während der Ausführung des Datenverarbeitungsprogramms auf dem lokalen Computer erforderlich macht.

[0054] Wenn der Nutzer bei einer erneuten Ausführung des Datenverarbeitungsprogramms eine Funktion aufruft, die die Essenzdatei **7** benötigt, ermittelt das Datenverarbeitungsprogramm durch Zugriff auf die Systemregistrierungsdatenbank, daß die Essenzdatei bereits auf dem lokalen Computer vorhanden ist. Aus der Registrierungsdatenbank kann das Transferarchiv ermittelt werden, in welchem die Essenzdatei enthalten ist. Für den Fall, daß mehrere Essenzdateien in einem gemeinsamen Transferarchiv gespeichert sind, enthält die Registrierungsdatenbank auch einen Eintrag über die Position der jeweiligen Essenzdatei in dem Transferarchiv. Es ist dann keine erneute Anforderung der Essenzdatei vom Server erforderlich.

[0055] Das Transferarchiv enthält im Archivkopf auch Angaben über die auszuführende Routine, mit welcher die Grunddatei und die Essenzdateien zu verknüpfen sind. Ebenso befindet sich im Transferarchiv der jeweilige Sitzungsschlüssel für die einzelnen Essenzdateien, der unter Anwendung des Private-Key des Nutzers entschlüsselt werden muß. Da die Essenzdatei selbst mit einem symmetrischen Verschlüsselungsverfahren verschlüsselt wurde kann die Entschlüsselung während der Ausführung des Datenverarbeitungsprogramm (on-the-fly) erfolgen, da bei symmetrischen Verschlüsselungsverfahren eine ausreichend schnelle Entschlüsselung durchgeführt werden kann, sobald der symmetrische Schlüssel entschlüsselt wurde.

[0056] Es ist noch festzuhalten, daß auf dem Server auch die nutzerspezifischen Daten gespeichert werden. Diese ermöglichen bei einem späteren Zugriff des Nutzers auf den Server eine sofortige Identifizierung des Nutzers sowie die Überprüfung der bereits an diesen Nutzer gesendeten Essenzdateien. Dadurch kann sichergestellt werden, daß der Nutzer bestimmte Essenzdateien nicht doppelt vom Server beziehen muß. Selbst wenn durch einen Datenverlust die beim Nutzer gespeicherten Essenzdateien beschädigt werden, kann auf diese Weise ein erneuter Download vom Server ermöglicht werden, ohne daß die bereits bezahlten Essenzdateien neu vom Nutzer bezahlt werden müssen. Zusätzlich kann der Server auch Informationen über ein Konto eines Nutzers verwalten, von welchem ein entsprechendes Guthaben abgebucht werden kann, wenn der Nutzer Essenzdateien erwirbt. Der Server hält außerdem alle für eine bestimmte Anwendung verfügbaren Essenzdateien (Plugins) unverschlüsselt bereit, die verschlüsselt und in ein nutzerspezifisches Transferarchiv verpackt werden, wenn der Nutzer diese Essenzdateien beziehen will.

[0057] In **Fig. 3** ist weiterhin der Verfahrensablauf dargestellt, wenn beim Nutzer der Private-Key, der für die Entschlüsselung der Essenzdateien erforderlich ist, nicht mehr zur Verfügung steht. Dieser Fall kann beispielsweise eintreten, wenn der Private-Key nicht auf einer Smart-Card gespeichert ist, sondern in Bezug auf bestimmte Hardwarekomponenten des lokalen Computers ermittelt wurde und eine Veränderung der Hardwarekomponenten erfolgte. Die dann notwendige Kommunikation zwischen dem lokalen Computer startet im Schritt **40**. Wenn im Schritt **41** bestimmt wurde, daß eine Verbindung mit dem Server hergestellt wird, wird ein neues Schlüsselpaar erstellt und im Schritt **42** wiederum die Verbindung zum Server) aufgebaut. Dabei wird der neue öffentliche Schlüssel übertragen. Im Schritt **43** wird der neue Private-Key des Nutzers/lokalen Computers überprüft. Sofern der Erfolg der Überprüfung im Schritt **44** festgestellt wird, wechselt der Server im Schritt **45** in einen Wiederanmeldedialog, wobei der Benutzername (Login) unveränderlich ist. Im Schritt **46** wird eine Passwortüberprüfung durch den Server ausgeführt. Das Passwort ist dem Nutzer aus der vom Server gesendeten E-Mail bekannt. Wenn das Passwort in Schritt **47** bestätigt wurde, kann im Schritt **48** ein neues Passwort ermittelt werden. Anschließend wird das Verfahren im Schritt **29** fortgesetzt, so daß der Nutzer weitere Essenzdateien vom Server erhalten kann.

[0058] Der Private-Key kann entweder auf der Smart-Card abgelegt sein oder in verschlüsselter und/oder versteckter Form auf dem Lokaldatenträger des lokalen Computers gespeichert sein.

[0059] Dazu eignet sich z. B. eine Anbindung des Private-Key an spezielle Hardware-Merkmale des lokalen Computers.

[0060] Eine Wiederanmeldung kann auch erforderlich sein, wenn der Nutzer des Datenverarbeitungsprogramms seinen Rechner neu installiert hat. Dazu kann der Nutzer im Schritt **25** in einen Wiederanmeldedialog verzweigen (Schritt **50**). Nachdem er seinen alten Login eingegeben hat, erfolgt im Schritt **51** die Überprüfung des Logins und des Passwortes. Nach einer Bestätigung im Schritt **52** kann im Schritt **48** ein neues Passwort ermittelt werden. Das Verfahren wird dann wie beschrieben in Schritt **29** fortgesetzt.

[0061] Schließlich ist in **Fig. 3** der notwendige Verfahrensablauf dargestellt, wenn ein dem Server bereits bekannter Nutzer bzw. lokaler Computer eine weitere Essenzdatei benötigt, beispielsweise um in dem lokal auszuführenden Datenverarbeitungsprogramm ein zusätzliches Level eines Computerspiels auszuführen. Die Wiederanmeldung des Nutzer für eine Übertragung einer weiteren Essenzdatei startet im Schritt **60**. Vor **60** wird erneut entschieden, ob die benötigte Essenzdatei vom Server bezogen werden soll. Nach einer Rückfrage im Schritt **61** wird im Schritt **62** ein Verbindung mit dem Server aufgebaut. Dabei überträgt der Pay-per-use-Client mit der POST-Methode die für die Verbindung erforderlichen Parameter, insbesondere den Benutzernamen (Login), Identifikationsdaten für das Datenverarbeitungsprogramm, Identifikationsdaten für die benötigte Essenzdatei und eine digitale Signatur. Die Signatur wird im Schritt **63** überprüft. Wenn die übertragenen Daten im Schritt **64** verifiziert werden konnten, gelangt der Nutzer sofort in den Auswahlbereich (E-Shop), wo er weitere Essenzdateien (Plugins) zum Download auswählen kann (Schritt **65**). Bei einer abgewandelten Ausführungsform kann dieser Schritt entfallen, wenn bereits durch die erste Datenübertragung festgelegt wird, welche Essenzdatei benötigt wird und damit eine weitere Auswahl nicht erforderlich ist.

[0062] Im Schritt **66** wird ein individuelles Transferarchiv erstellt, welches die angeforderten Essenzdateien enthält. Im Schritt **67** wird das Transferarchiv auf den lokalen Computer übertragen, so daß dort in Schritt **68** in der beschriebenen Weise das Transferarchiv überprüft und gespeichert werden kann.

[0063] Sofern ein an sich beim Server bekannter Nutzer für ein anderes Datenverarbeitungsprogramm (z. B. neues Computerspiel) Essenzdateien erwerben will, kann er auf dem Server als alter Nutzer behandelt werden, dem für dieses neue Datenverarbeitungsprogramm ein neues Passwort zugeteilt wird, so daß das entsprechende Verfahren beginnend im Schritt **20** abläuft.

[0064] Das beschriebene Verfahren zur Distribution von multimedialen Datenmengen und das Datenverarbeitungssystem können auch für andere multimediale Datenmengen verwendet werden. Beispielsweise wäre es denkbar, bestimmte Detailinformationen aus elektronischen Landkarten in diese Essenzdateien zu verlagern, die der Nutzer erst bei Bedarf beziehen kann.

Patentansprüche

1. Verfahren zur Verfügbarmachung von multimedialen Datenmengen für einen Nutzer, die folgenden Schritte umfassend:
   - Bereitstellung einer Essenzdatei (**7**) auf einem entfernten Server (**3**), die wenigstens einen Teil der multimedialen Datenmenge umfaßt;
   - Verschlüsseln der Essenzdatei;
   - Übertragung der verschlüsselten Essenzdatei zu einem lokalen Computer (**1**), auf welchen der Nutzer Zugriff hat;

     Speicherung der verschlüsselten Essenzdatei auf einem Lokaldatenträger des lokalen Computers;
   - Entschlüsselung der Essenzdatei während der Ausführung eines Datenverarbeitungsprogramms (**2**) auf dem lokalen Computer;
   - Wiedergabe der multimedialen Datenmenge über ein Ausgabegerät, während der Ausführung des Datenverarbeitungsprogramms auf dem lokalen Computer.

2. Verfahren nach Anspruch 1, wobei die Essenzdatei (**7**) die vollständige multimediale Datenmenge enthält.

3. Verfahren nach Anspruch 1, wobei die Essenzdatei nicht die gesamte multimediale Datenmenge umfaßt, die folgenden Schritte umfassend:
   - Aufteilung der multimedialen Datenmenge in eine Grunddatei (**6**) und eine Essenzdatei (**7**), gemäß einem vorgegebenen Teilungsalgorithmus;
   - Übertragung der Grunddatei und der Essenzdatei;
   - gemeinsame Speicherung der Grunddatei und der verschlüsselten Essenzdatei auf dem Lokaldatenträger eines lokalen Computers (**1**);
   - Verknüpfung der Grunddatei und der Essenzdatei zur multimedialen Datenmenge während der Ausführung des Datenverarbeitungsprogrammes (**2**) auf dem lokalen Computer, nach der Entschlüsselung der Essenzdatei.

4. Verfahren nach Anspruch 3, wobei der Schritt der Aufteilung der Datenmenge das Extrahieren wesentlicher Teile der Datenmenge in die Essenzdatei umfaßt, so daß die Grunddatei ohne die Essenzdatei nicht mehr oder nur noch mit verminderter Qualität bzw. verringertem Funktionsumfang am Ausgabegerät wiedergegeben werden kann.

5. Verfahren nach Anspruch 4, wobei aus der Datenmenge zumindest Teile der Dateikopfinformation (Header), die für die Interpretation der restlichen Daten notwendig sind, extrahiert und in die Essenzdatei eingebunden werden.

6. Verfahren nach Anspruch 4, wobei stereophone Audiodaten vor ihrer Codierung in einen Monodatenstrom (Summensignal: rechter plus linker Kanal) und einen Differenzdatenstrom (rechter minus linker Kanal) aufgeteilt werden, und wobei der Rechts-Links-Datenstrom die Essenzdatei bildet.

7. Verfahren nach Anspruch 4, wobei die in die Essenzdatei zu verlagernden Daten erst nach einer Komprimierung der multimedialen Datenmenge aus der komprimierten Datenmenge extrahiert werden.

8. Verfahren nach einem der Ansprüche 1 bis 7, wobei die Verschlüsselung der Essenzdatei mit einem symmetrischen Verschlüsselungsalgorithmus erfolgt, und wobei der dafür verwendete sitzungsspezifische Schlüssel (Session-Key) seinerseits durch einen asymmetrischen Verschlüsselungsalgorithmus verschlüsselt wird.

9. Verfahren nach Anspruch 8, weiterhin die folgenden Schritte umfassend:
   - Ermittlung einer Zufallszahl als sitzungsspezifischen Schlüssel (Session-Key);
   - Verschlüsselung der Essenzdatei (**7**) mit dem sitzungsspezifischen Schlüssel unter Anwendung des symmetrischen Verschlüsselungsalgorithmus;
   - Verschlüsselung des sitzungsspezifischen Schlüssels mit dem öffentlichen Schlüssel eines dem Nutzer gehörenden Schlüsselpaars unter Anwendung des asymmetrischen Verschlüsselungsalgorithmus;

     nochmalige Verschlüsselung des asymmetrisch verschlüsselten Sitzungsschlüssels mit einem nutzerspezifischen Schlüssel (Passwort) unter Anwendung eines symmetrischen Verschlüsselungsalgorithmus.

10. Verfahren nach Anspruch 9, wobei der Schritt der Übertragung der verschlüsselten Essenzdatei das Erzeugen eines Transferarchivs umfaßt, welches übertragen wird und die folgenden Komponenten enthält:
   - die verschlüsselte Essenzdatei;
   - eine Serversignatur, die die Identifizierung des entfernten Servers ermöglicht;
   - einen Prüfabschnitt, der eine Fehlerprüfung des Transferarchivs ermöglicht;
   - einen Datenidentifikationsabschnitt;
   - einen Nutzerdatenabschnitt, der die Identifizierung des Nutzers ermöglicht;
   - den verschlüsselten Sitzungsschlüssel zur Entschlüsselung der Essenzdatei.

11. Verfahren nach einem der Ansprüche 1 bis 10, wobei nach der Speicherung der verschlüsselten Essenzdatei auf dem Datenträger des lokalen Computers Einträge in einer Systemregistrierungsdatenbank des lokalen Computers generiert werden, aus denen das Datenverarbeitungsprogramm während des Ablaufs die Speicherposition der Essenzdatei ermitteln kann.

12. Verfahren nach Anspruch 9, soweit er auf Anspruch 3 rückbezogen ist, wobei das Transferarchiv mit der verschlüsselten Essenzdatei über eine Online-Verbindung vom Server zum lokalen Computer übertragen wird, und wobei weiterhin die folgenden Schritte ausgeführt werden:
   - Übertragung einer Anforderung der Essenzdatei vom lokalen Computer zum Server;
   - Überprüfung der Berechtigung des Nutzers

zum Empfang der Essenzdatei;
- Erzeugung des Transferarchivs auf dem Server;
- Ausführung einer Finanztransaktion zur Bezahlung der Essenzdatei, bevor das Transferarchiv auf den lokalen Computer übertragen wurde.

13. Verfahren nach einem der Ansprüche 1 bis 12, wobei auf dem Server Daten gespeichert werden, die eine eindeutige Identifikation des lokalen Computers, der an diesen übertragenen Essenzdateien und ggf. des jeweiligen Nutzers ermöglichen.

14. Datenverarbeitungssystem zur Verfügbarmachung von multimedialen Datenmengen, umfassend:
- einen entfernten Server (3) mit einem Serverdatenträger;
- einen lokalen Computer (1) mit einem Lokaldatenträger und einem Ausgabegerät (14) für multimediale Daten, der ein die multimedialen Daten verwendendes Datenverarbeitungsprogramm ausführt;
- einer zumindest zeitweiligen Datenübertragungsverbindung (4) zwischen dem Server und dem lokalen Computer;

gekennzeichnet durch
- eine Verschlüsselungseinrichtung, die eine Essenzdatei verschlüsselt, welche wenigstens einen Teil der multimedialen Datenmenge umfaßt;
- eine Archiverzeugungseinrichtung des Servers, die die verschlüsselte Essenzdatei mit nutzerspezifischen Daten zu einem Transferarchiv kombiniert und diese für die Online-Übertragung über die Datenübertragungsverbindung bereitstellt;
- eine Aktualisierungseinrichtung des lokalen Computers, die Einträge in eine Systemregistrierungsdatenbank des lokalen Computers vornimmt, welche die Inhalte des übertragenen Transferarchivs angeben;
- eine Entschlüsselungseinrichtung des lokalen Computers, welche während der Ausführung eines Datenverarbeitungsprogramms (2) die Essenzdatei entschlüsselt; und
- eine Ausgabeeinrichtung, welche die multimediale Datenmenge an das Ausgabegerät ausgibt.

15. Datenverarbeitungssystem nach Anspruch 14, weiterhin gekennzeichnet, durch
- eine Dateiteilungseinrichtung, die die multimediale Datenmenge in eine Grunddatei (6) und eine Essenzdatei (7) aufteilt; und
- eine Kombinationseinrichtung des lokalen Computers, welche die entschlüsselte Essenzdatei gemäß einem Kombinationsalgorithmus mit der Grunddatei zu der multimedialen Datenmenge verknüpft.

16. Datenverarbeitungssystem nach Anspruch 14 oder 15, dadurch gekennzeichnet, daß:
- die Verschlüsselungseinrichtung die Essenzdatei (7) mit einem sitzungsspezifischen Schlüssel (Session-Key) durch einen symmetrischen Verschlüsselungsalgorithmus verschlüsselt;
- die Archiverzeugungseinrichtung den sitzungsspezifischen Schlüssel mit dem öffentlichen Schlüssel eines Schlüsselpaars des Nutzers durch einen asymmetrischen Verschlüsselungsalgorithmus verschlüsselt und dem Transferarchiv hinzufügt;
- die Entschlüsselungseinrichtung mit Hilfe des privaten Schlüssels des Schlüsselpaars des Nutzers den sitzungsspezifischen Schlüssel entschlüsselt und mit diesem die Essenzdatei entschlüsselt.

17. Datenverarbeitungssystem nach Anspruch 16, dadurch gekennzeichnet, daß die Archiverzeugungseinrichtung den sitzungsspezifischen Schlüssel (Session-Key) weiterhin mit einem nutzerspezifischen Schlüssel (Passwort) unter Anwendung eines symmetrischen Verschlüsselungsalgorithmus verschlüsselt.

18. Datenverarbeitungssystem nach Anspruch 16 oder 17, dadurch gekennzeichnet, daß der private Schlüssel des Nutzers in einer verschlüsselten Form auf dem lokalne Computer gespeichert ist, wobei der private Schlüssel unter Anwendung eines symmetrischen Verschlüsselungsverfahrens mit einem nutzerspezifischen Schlüssel verschlüsselt ist, und wobei der nutzerspezifische Schlüssel gemäß einem vorgegebenen Generierungsalgorithmus aus gerätetechnischen Merkmalen (Hardware) und der Konfigurationsmerkmalen des lokalen Computers erzeugt wird.

19. Datenverarbeitungssystem nach Anspruch 16 oder 17, dadurch gekennzeichnet, daß der lokale Computer eine Smart-Card-Einrichtung mit einer Smart-Card umfaßt, auf welcher der private Schlüssel des Nutzers abgelegt ist, wobei die Entschlüsselungseinrichtung zur Entschlüsselung des symmetrischen Schlüssels auf die Smart-Card zugreift.

20. Datenverarbeitungssystem nach einem der Ansprüche 14 bis 19, dadurch gekennzeichnet, daß die einzelnen Einrichtungen durch Softwareroutinen realisiert sind.

21. Computer, der als Server konfiguriert ist und eine Datenübertragungsverbindung zu anderen Computern aufbauen kann, umfassend:
- eine Dateiteilungseinrichtung, die eine multimediale Datenmenge in eine Grunddatei und eine Essenzdatei aufteilt;
- eine Verschlüsselungseinrichtung, die die Essenzdatei verschlüsselt;
- eine Archiverzeugungseinrichtung, die die verschlüsselte Essenzdatei mit spezifischen Daten eines verbundenen Computers zu einem Transferarchiv kombiniert und diese für eine Online-Übertragung über die Datenübertragungsverbindung bereitstellt;
- eine Nutzerdatenbank, in der nutzerspezifische Daten einschließlich der Merkmale der übertragenen Transferarchive gespeichert werden.

22. Computer nach Anspruch 21, dadurch gekennzeichnet, daß weiterhin eine Schlüsselgenerierungseinrichtung vorhanden ist, die einen zufälligen sitzungsspezifischen Schlüssel (Session-Key) zur Verschlüsselung der Essenzdatei erzeugt.

23. Computer, der als lokaler Computer konfiguriert ist, eine Datenübertragungsverbindung zu einem Server aufbauen kann, wobei er die Funktion des Clients übernimmt, und eine Aktualisierungseinrichtung umfaßt, welche Einträge in eine Systemregistrierungsdatenbank vornimmt, welche die Inhalte eines vom Server übertragenen Transferarchivs angeben, wobei auf dem lokalen Computer ein Datenverarbeitungsprogramm ausführbar ist, welches während seiner Ausführung eine im Transferarchiv enthaltene Essenzdatei entschlüsselt, die entschlüsselte Essenzdatei gemäß einem im Transferarchiv bestimmten Kombinationsalgorithmus mit einer lokal gespeicherten Grunddatei zu einer multimedialen Datenmenge verknüpft und diese an ein Ausgabegerät ausgibt.

Hierzu 5 Seite(n) Zeichnungen

Nummer: DE 100 59 230 A1
Int. Cl.[7]: H 04 L 9/32
Offenlegungstag: 13. Juni 2002

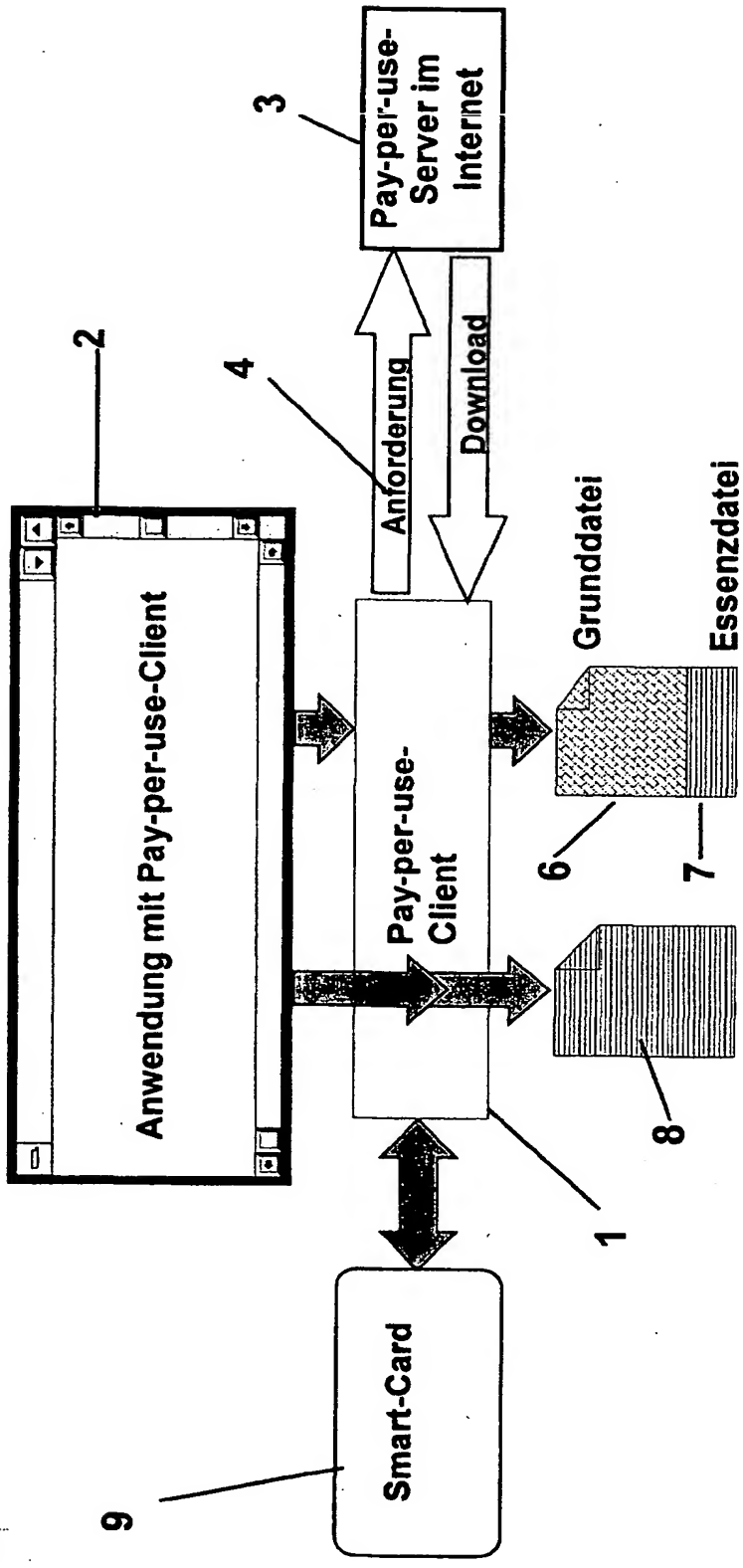


Fig. 1

Nummer: DE 100 59 230 A1
Int. Cl.[7]: H 04 L 9/32
Offenlegungstag: 13. Juni 2002

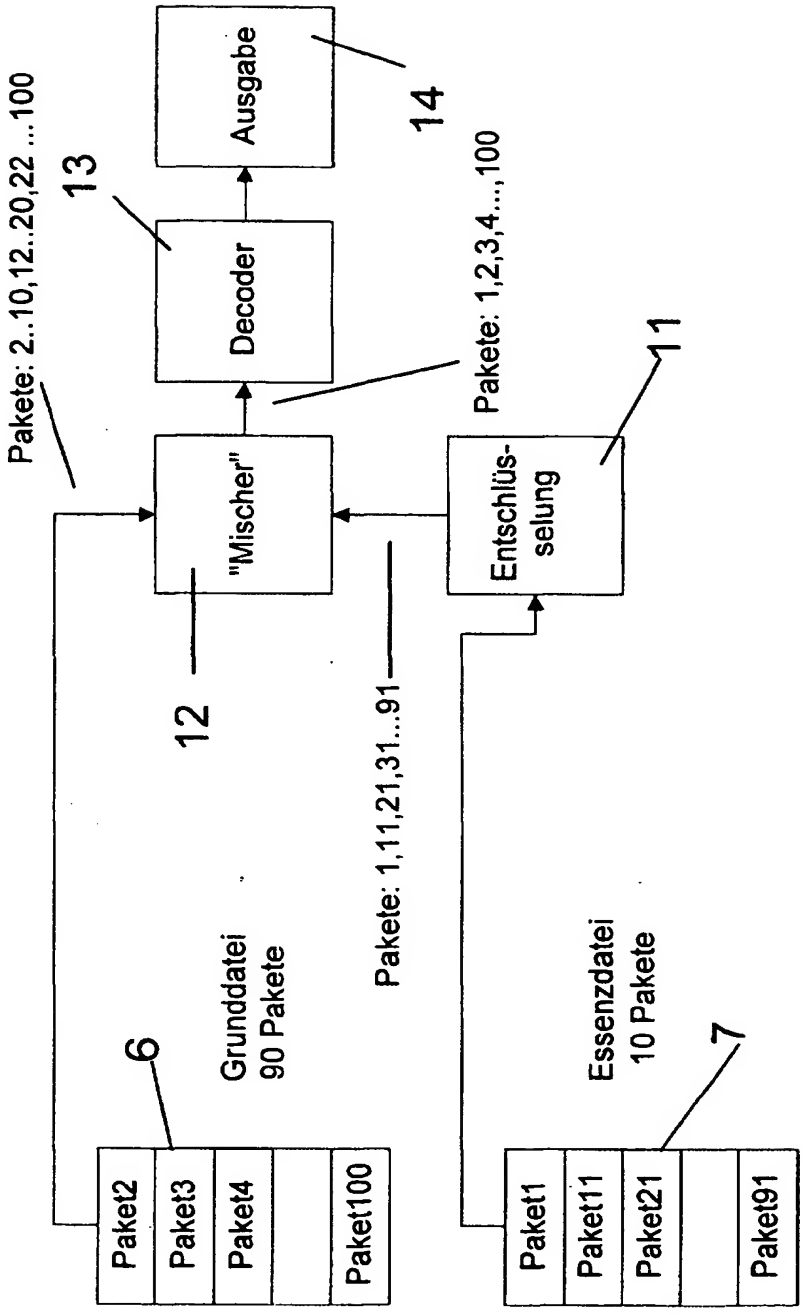
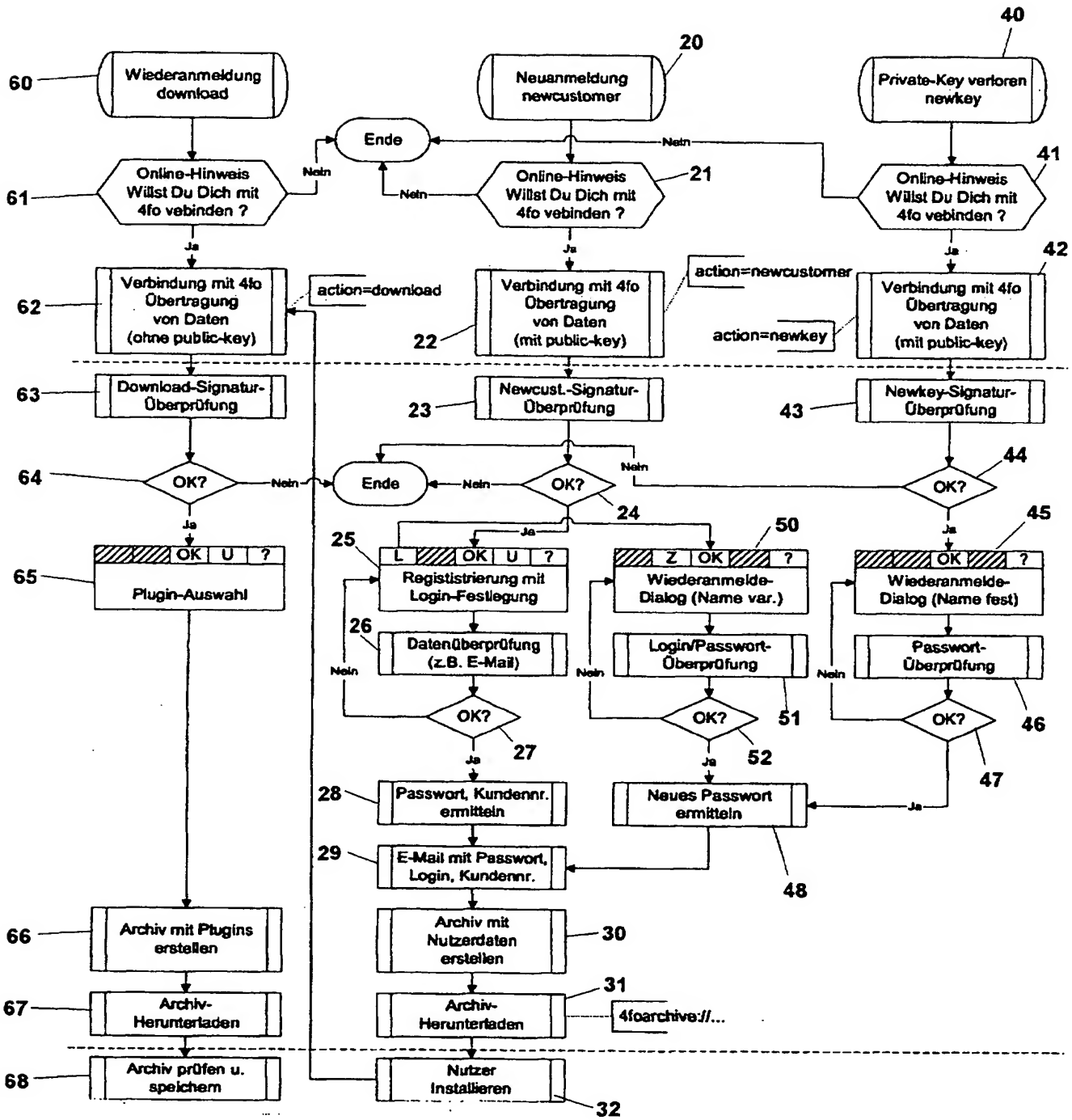


Fig. 2

Nummer: DE 100 59 230 A1
Int. Cl.[7]: H 04 L 9/32
Offenlegungstag: 13. Juni 2002

Fig. 3



Nummer: DE 100 59 230 A1
Int. Cl.[7]: H 04 L 9/32
Offenlegungstag: 13. Juni 2002

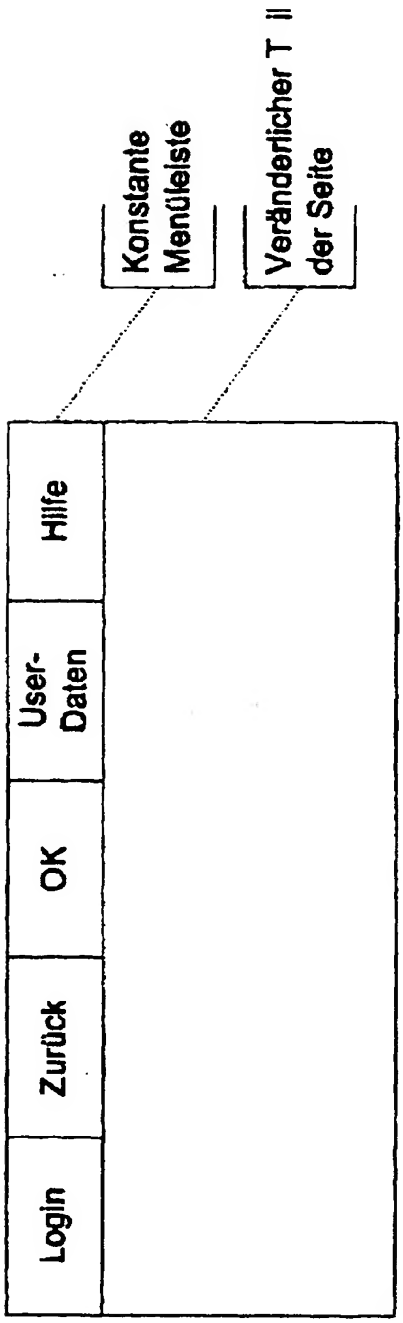


Fig. 4

Nummer: DE 100 59 230 A1
Int. Cl.[7]: H 04 L 9/32
Offenlegungstag: 13. Juni 2002

Fig. 5 Struktur des Transferarchivs

```
<4FOArchive> ::= <ArchiveHeader> <Plugins> <serverSignature> <crc32>
<ArchiveHeader> ::=  <archiveVersion> <gameID> <archiveID> <createDate>
                     <customerNo> <customerName> <encipheredKey> <nmbPlugins>
                     <PluginInfo> {<PluginInfo>}
<Plugins> ::= <sessionID> <dataLength> {<dataLength>} <PluginData> {<PluginData>}
<PluginInfo> ::= <filenameLength> <originalFilename> <fileTypeID>

<serverSignature> ::=  172 Bytes (Base64-Codierung)
                       Hash über <ArchiveHeader> und <Plugins>, signiert mit
                       spielespezifischen private key des Servers (1024 Bit)
<crc32> ::=          32 Bit Integer (unsigned)
                     CRC32 über <ArchiveHeader>, <Plugins> und <serverSignature>
<archiveVersion> ::= 32 Bit Integer (unsigned)        0x2 = aktuelle Struktur
<gameID> ::=         32 Bit Integer (unsigned)
                     bezeichnet das Spiel (mit Version) eindeutig
                     00000101x0 = Demonstrator in der ersten Version
                     00000102x0 = Demostrator in der Version 2
                     00000201x0 = Demospiel in der ersten Version
<archiveID> ::=      32 Bit Integer (unsigned)
                     bezeichnet das Archiv (mit Version) eindeutig,
                     protokolliert fortlaufend die Archive, die für eine GameID
                     ausgegeben wurden
<createDate> ::=     8 Bytes
                     Erstellungsdatum, Format: z.B. 01202000 für den 20. Jan. 2000
<customerNo> ::=        32 Bit Integer (unsigned)        Kundennummer
<customerName> ::=      33 Bytes (= max. 32 Zeichen + Rest #0)
                        Login-Name/Nickname (Ansi-Zeichensatz, Buchstaben
                        (international) und [0-9]_.-<Blank>)
<encipheredKey> ::=     172 Bytes (Base64-Codierung)
                        128 Bit Zufallszahl = Symmetrischer Schlüssel
                        kann später vergrößert werden, mit dem Passwort (64Bit
                        symmetrisch) und dem public key (1024 asymmetrisch) des
                        Kunden verschlüsselt
<nmbPlugins> ::= 32 Bit Integer (unsigned)      Anzahl der enthaltenen Plugins
<filenameLength> ::= 32 Bit Integer (unsigned) bezeichnet die Länge des Filenamens
<originalFilename> ::=    <filenameLength> Bytes
                          Dateiname plus relativer Pfad (Windows-Style)
<fileTypeID> ::=     32 Bit Integer (unsigned)     Plugintyp
                     z.B.:0x1 = Datenplugins (Dateianfang)
<sessionID> ::=      256 Bytes (Base64-Codierung)       Session-ID vom Server
<dataLength> ::=     32 Bit Integer (unsigned)          pluginRawData.size

<PluginData> ::=        <pluginRawData> <paddingBytes> <pluginDataCrc32>
                        (Blowfish-Verschlüsselung)
<pluginRawData> ::=     <dataLength> Bytes
                        Plugin (ergibt zusammen mit der Restdatei das Originalfile)
<paddingBytes> ::=      Zufalls-Bytes
                        um <PluginData>.size auf nächstes 8-Bytes-Fenster zu erhöhen
<pluginDataCrc32> ::=   32 Bit Integer (unsigned)     CRC32 über <pluginRawData>
```

Strings werden in Feldern konstanter Länge abgespeichert. Nicht belegte Zeichen werden mit binären Nullen gefüllt.

4FOArchiv-Filename:

Der Dateiname des Transfer-Archives wird aus der SessionID des Pay-per-use-Server direkt abgeleitet: <die letzten 32 Zeichen der sessionID>.4fo